

# Utilizing Oracle Solaris Containers with Oracle Database

Björn Rost

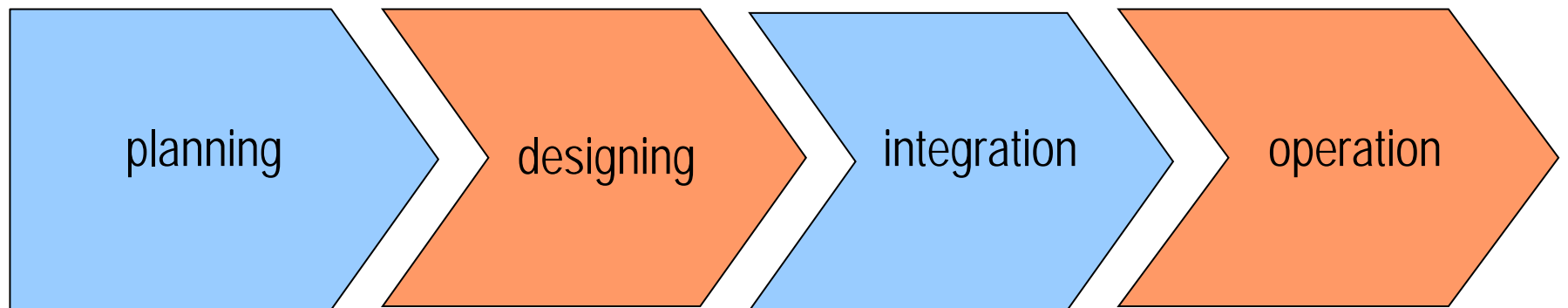
# about us

- Software Production company founded 2001
  - mostly J2EE
    - logistics
    - telco
    - media and publishing
- customers expect full lifecycle support
  - hardware resale
  - datacenter and network
  - 3<sup>rd</sup> party software



# project lifecycle

- consulting
- specification
- documentation
- feasibility studies
- J2EE
- php
- database
- Hardware
- SW-Licenses
- Installation
- Benchmarking
- Tuning
- Monitoring
- Support
- Updates
- Backups



**blau**

Mobilfunk GmbH

**M EEDIA**

**portrix**  
systems

**ANRUFDIREKT**  
predictive dialing by call

**VR** VIRTUALREEL

**e+p**

**E**  
EDEKA

**SevenSeas** International Publishing Agency

**LichtBlick**  
die Zukunft der Energie

**apo-rot**<sup>®</sup>  
Ihre VersandApotheke

**gk** Klett

**EatSmarter!**

**Kolle Rebbe**

**LFI**  
LEICA FOTOGRAFIE INTERNATIONAL

**Commercial**  
**Film**  
**Service**

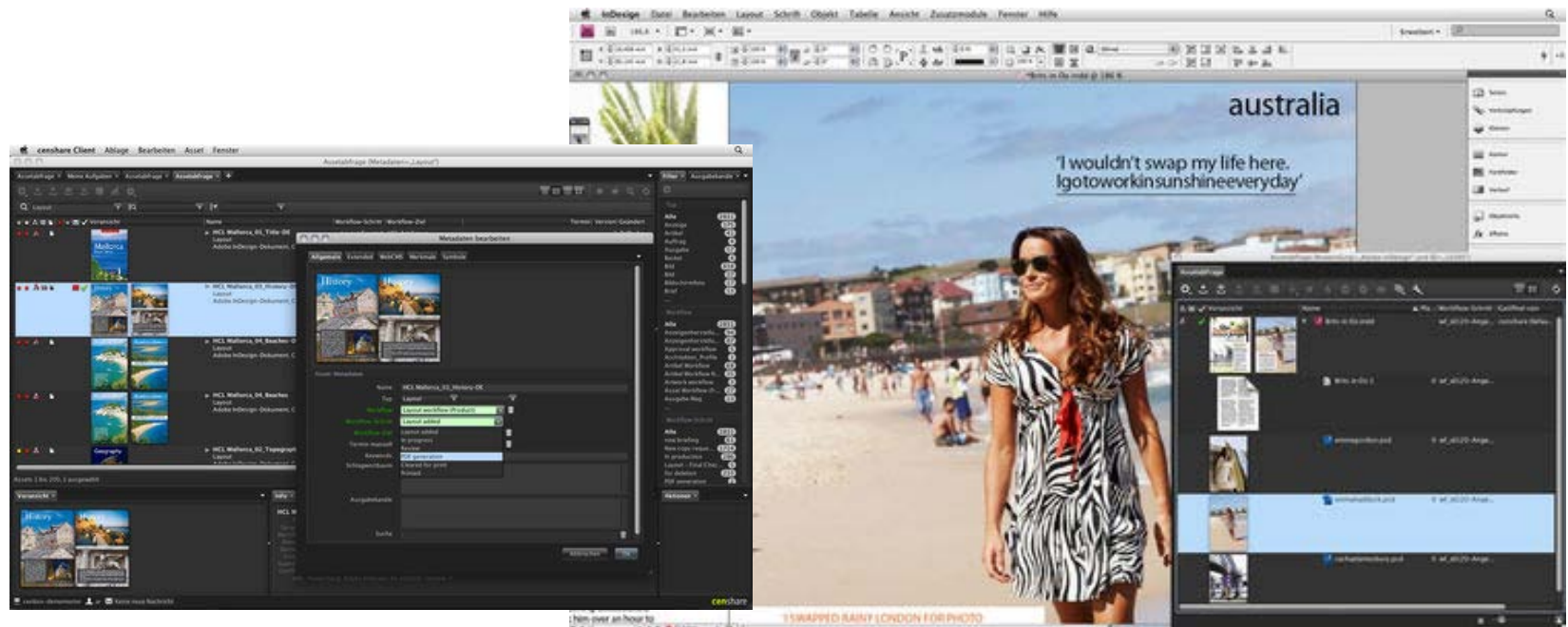
**JUNG v. MATT**

# 7-seas

- global publishing system
  - document management
  - translation
  - timing control



**cen**share  
GLOBAL PUBLISHING



# 7-seas

- client/server based system
  - JAVA application Server
  - Oracle Database
  - Apache Tomcat Webserver



# the challenge

- deploy and run this multi-tiered system
  - cost-effectively
    - 3<sup>rd</sup> party licenses
    - hardware
    - operating costs
  - but stay flexible to scale with demand
  - and have a plan for recovery

# Why virtualize?

- Consolidation saves resources, money
- Flexible deployment and scaling
- Oracle hard partitioning saves licensing
- availability (better to have one SPOF than many)
- poor-man's flashback
- poor-man's compression
- cheap and easy cloning



# Consolidate

- many systems are running at low utilization
  - you might not even know the impact on subsystems at launch
- systems get more and faster cores all the time
  - but not all workflows do, too
- energy costs increasing

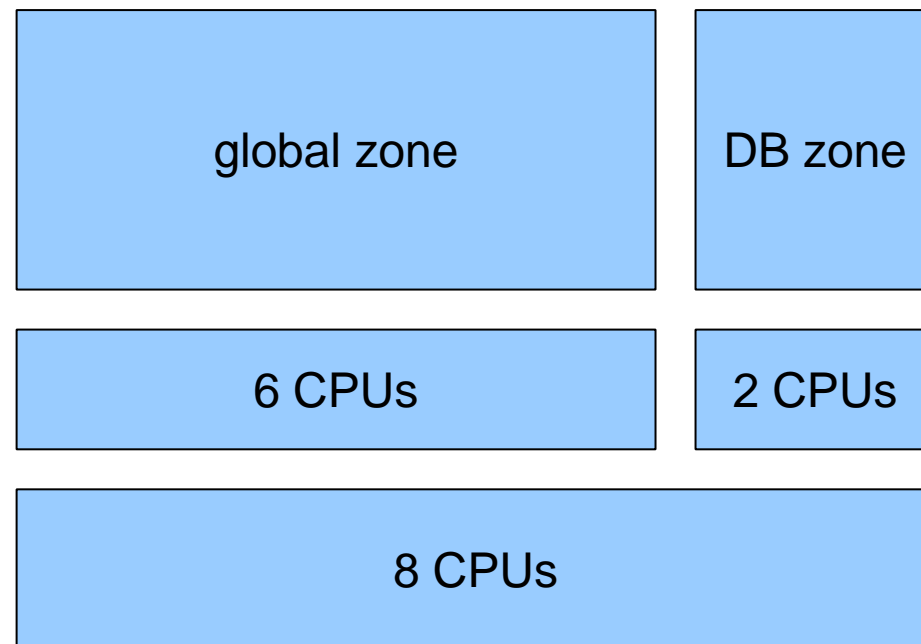


# Flexible Deployment

- deploy a system once
- in a virtual environment
- then, move it across servers to scale
- start smart, scale hard
  - initial deployment on single box with multiple VMs
  - scale out by distributing VMs across nodes when needed
- easy cloning of environments
- in case of HW failure restore VM to another box

# Oracle License Partitioning

- Only pay for the CPUs/Cores you actually use
- requires 'hard' partitioning
- grow/scale easily – buy new license and enable cores



# Solutions

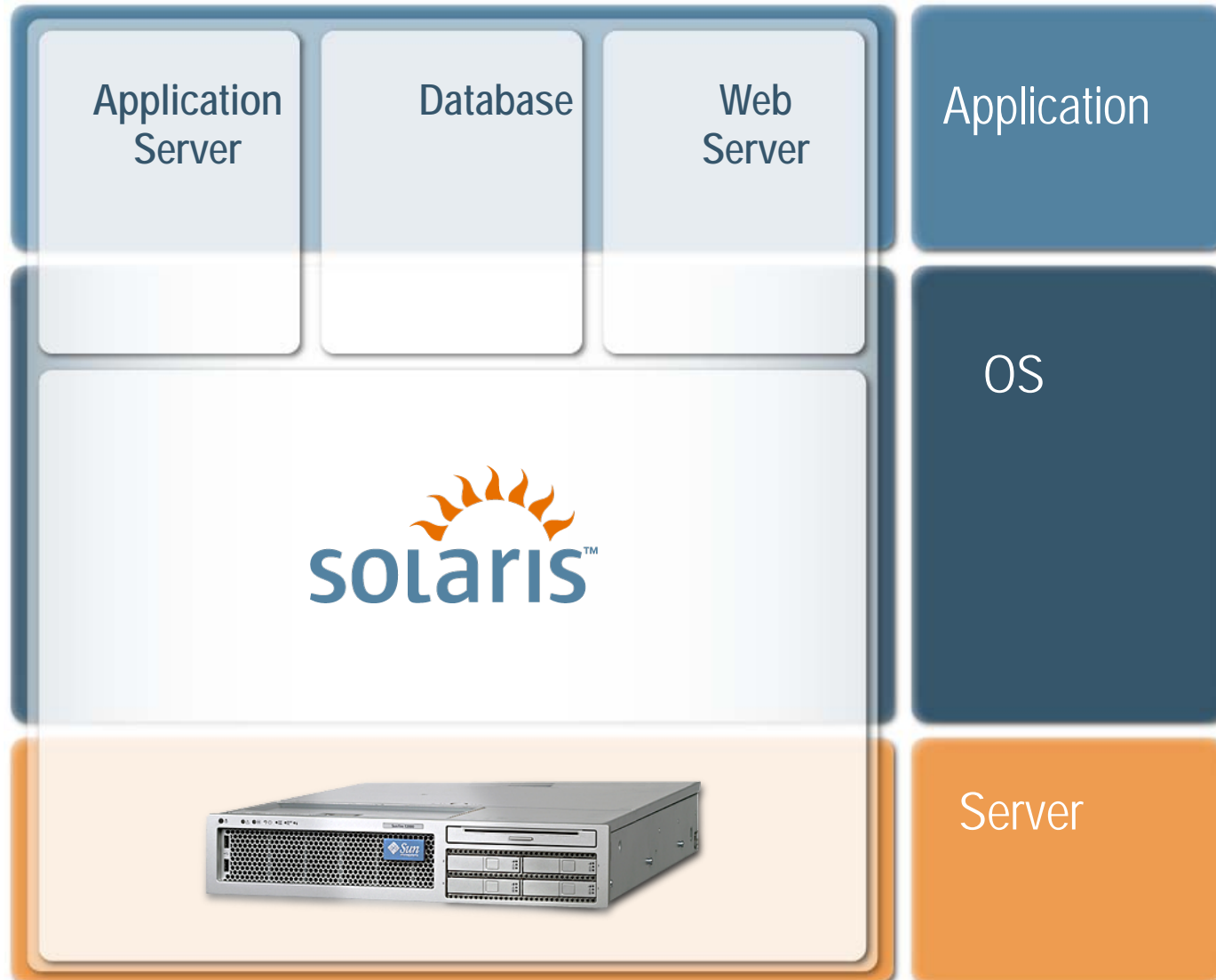
- VMware
  - possible issues with support and licensing
- LDom, vPar, nPar, LPAR, DSD
  - vendor- and hardware specific / non-x86
- Oracle VM
  - XEN hypervisor, typically used with OEL as guest OS
  - great for large-scale deployments
- Oracle Solaris Containers
  - native to Solaris, no extra software needed

# Containers

- Virtualization built into kernel – no installation, no costs
  - Resource Management
  - provides namespace, security and fault isolation
- runs on Solaris x86/x64 and SPARC
- no/minimal overhead
  - no hypervisor necessary
  - no special hardware needed
- mobility: detach/attach and clone
- Oracle supported and hard partitioning possible



# Oracle Solaris Zones



# container config

- global zone – this is the default/main OS instance
- local zone – these are the virtualized environments
  - own IP (and possibly interface)
  - sparse root zone
    - parts of directory structure (/usr, /lib ...) inherited from global zone
    - read-only in zone, saves storage space
  - full root zone
    - full directory structure is copied to zone
    - better control over patches in zone

# advanced config

- shared/private IP
- mapping devices / filesystems
- resource capping – container
- migration between servers
- branded zones (linux or solaris 9)



# step-by-step installation

- install Solaris 10 or 11
  - zones, resource management and zfs already included
- create zfs and zone(s)
- install oracle into zone
- PLEASE TRY THIS AT HOME!

# Demo

- zonecfg
- installation

```
root@hermes:~# zonecfg -z orallg
orallg: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:orallg> create
zonecfg:orallg> set zonepath=/zp03/zones/orallg
zonecfg:orallg> set autoboot=true
zonecfg:orallg> add capped-cpu
zonecfg:orallg:capped-cpu> set ncpus=1
zonecfg:orallg:capped-cpu> end
zonecfg:orallg> add net
zonecfg:orallg:net> set address=192.168.42.79
zonecfg:orallg:net> set physical=e1000g0
zonecfg:orallg:net> end
zonecfg:orallg>
```

```
zonecfg:orallg> info
zonename: orallg
zonepath: /zp03/zones/orallg
brand: native
autoboot: true
bootargs:
pool:
limitpriv:
scheduling-class:
ip-type: shared
inherit-pkg-dir:
    dir: /lib
inherit-pkg-dir:
    dir: /platform
inherit-pkg-dir:
    dir: /sbin
inherit-pkg-dir:
    dir: /usr
net:
    address: 192.168.42.79
    physical: e1000g0
    defrouter not specified
capped-cpu:
    [ncpus: 1.00]
rctl:
    name: zone.cpu-cap
    value: (priv=privileged,limit=100,action=deny)
zonecfg:orallg> verify
zonecfg:orallg> commit
zonecfg:orallg> exit
```

```
root@hermes:~# zoneadm -z orallg verify  
WARNING: /zp03/zones/orallg does not exist, so it could  
not be verified.  
When 'zoneadm install' is run, 'install' will try to  
create  
/zp03/zones/orallg, and 'verify' will be tried again,  
but the 'verify' may fail if:  
the parent directory of /zp03/zones/orallg is group- or  
other-writable  
or  
/zp03/zones/orallg overlaps with any other installed  
zones.
```

```
root@hermes:~# zoneadm -z orallg install
```

```
A ZFS file system has been created for this zone.
```

```
Preparing to install zone <orallg>.
```

```
Creating list of files to copy from the global zone.
```

```
Copying <147436> files to the zone.
```

```
Initializing zone product registry.
```

```
Determining zone package initialization order.
```

```
Preparing to initialize <1504> packages on the zone.
```

```
Initialized <1504> packages on zone.
```

```
Zone <orallg> is initialized.
```

```
The file
```

```
</zp03/zones/orallg/root/var/sadm/system/logs/install_log>
```

```
contains a log of the zone installation.
```

```
root@hermes:~# zoneadm -z orallg boot  
root@hermes:~# zlogin -C orallg
```


now, you get the 'normal' postinstall questions.  
language, hostname, root-pw, timezone ...

# DEMO

- zoneadm boot
- prstat -Z
- ps -Z



# ZFS

- ships for free with solaris – no installation needed
- new and innovative FS, Volume Manager
  - snapshots/clones
  - compression
  - quotas
  - raid
  - bootable  new
  - share with NFS or iSCSI
- enhances manageability in combination with zones

# Oracle datafiles on ZFS

- might not perform as well as ASM
- but taking snapshots of the whole database is amazing
  - fast and deduplicated DB cloning
  - poor-man's flashback database
  - poor-man's compression
  - backup/recovery

# demo

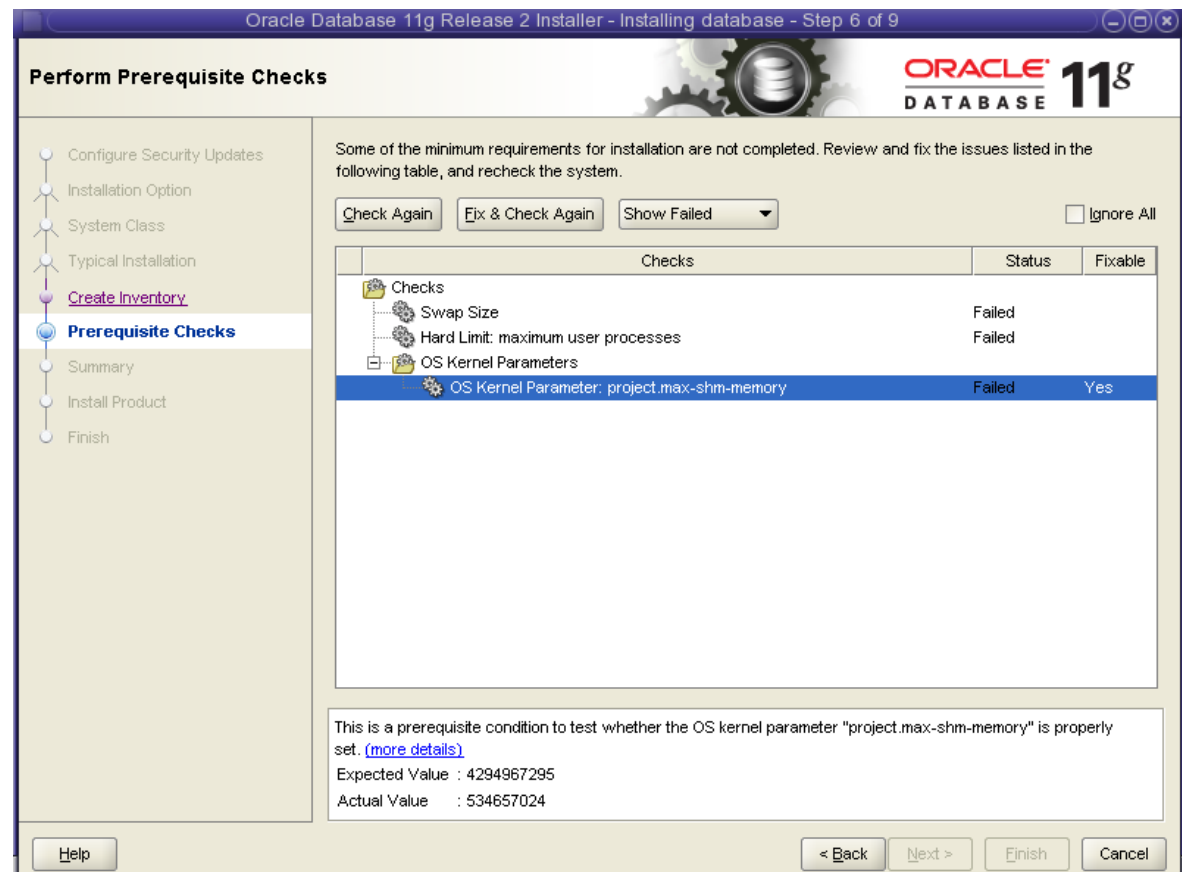
- snapshot
- cloning
- poor man's flashback

# licenses needed

- Solaris (already included in OPS with SUN hardware)
  - enterprise OS
  - zfs file system and volume manager
  - container virtualization
  - dTrace profiler
- Oracle
  - only for cores actually used

# known issues

- performance management
- limitpriv
  - proc\_prioctl
  - proc\_lock\_memory
- limit netservices
- kernel parameters
- ro /usr



# Solaris 11

- Solaris 10 branded zones
- no more sparse root zones
  - but IPS will change deployments anyway
  - even more flexibility!
- crossbow – network virtualization

Thank you!

questions?